Present and Future Computing Requirements for **Performance Optimization of Scientific Applications:** *MPAS-Ocean & the HipGISAXS Suite*

Abhinav Sarje

Computational Research Division

Lawrence Berkeley National Laboratory

NERSC ASCR Requirements for 2017
January 15, 2014
LBNL

1. Case Studies

SUPER: **SciDAC Institute for *Sus*tained *P*erformance, *E*nergy, and *R*esilience**

- *Goal*: Ensure that DOE's computational scientists can exploit the emerging generation of HPC systems.
- *MPAS-Ocean* is an earth-system simulation code for modeling the oceans in climate and weather studies.
- *PI*: Robert Lucas (USC).
- *PI (LBNL)*: Leonid Oliker.

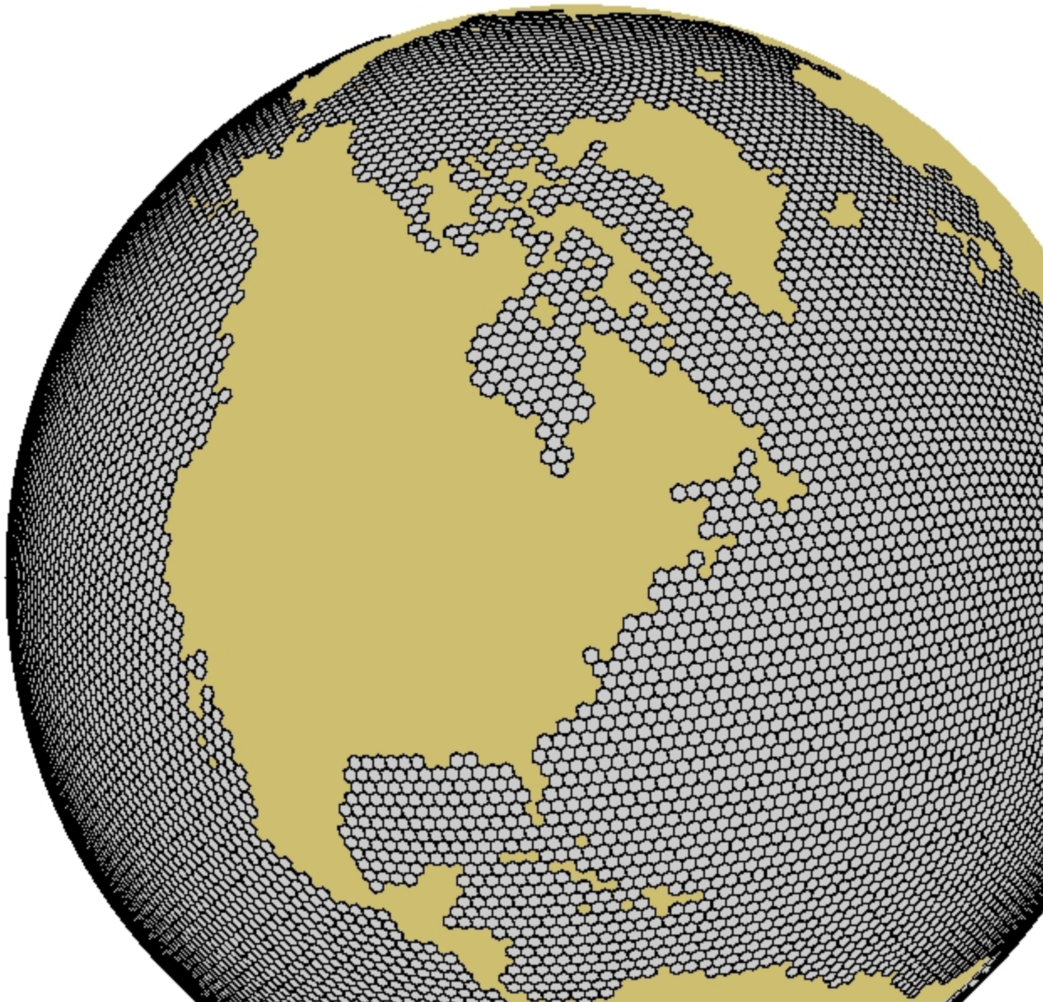
Next-Generation High-Performance Computing in X-Ray Science.

- *Goal*: Enable high-throughput X-Ray Scattering data analysis through massively parallel HPC systems.
- The *HipGISAXS Suite* is an X-Ray Scattering simulation and modeling code for micro/nanostructural studies in materials research.
- *PI*: Xiaoye Li (LBNL).

2. Performance Analysis and Optimization of MPAS-Ocean

- *MPAS* (Model for Prediction Across Scales)-*Ocean* developed at Los Alamos National Lab.

- Modeling and time-series simulations of Earth's oceans over an unstructured and multi-scale mesh discretization on a sphere.



2. Performance Analysis and Optimization of MPAS-Ocean

- Performance improvements through
  - better exploitation of data locality,
  - data reorganization, and
  - optimal data partitioning to minimize data transfers/communication;

- Hierarchical parallelization on multicores and exploitation of emerging architectures (Intel MICs, GPUs).

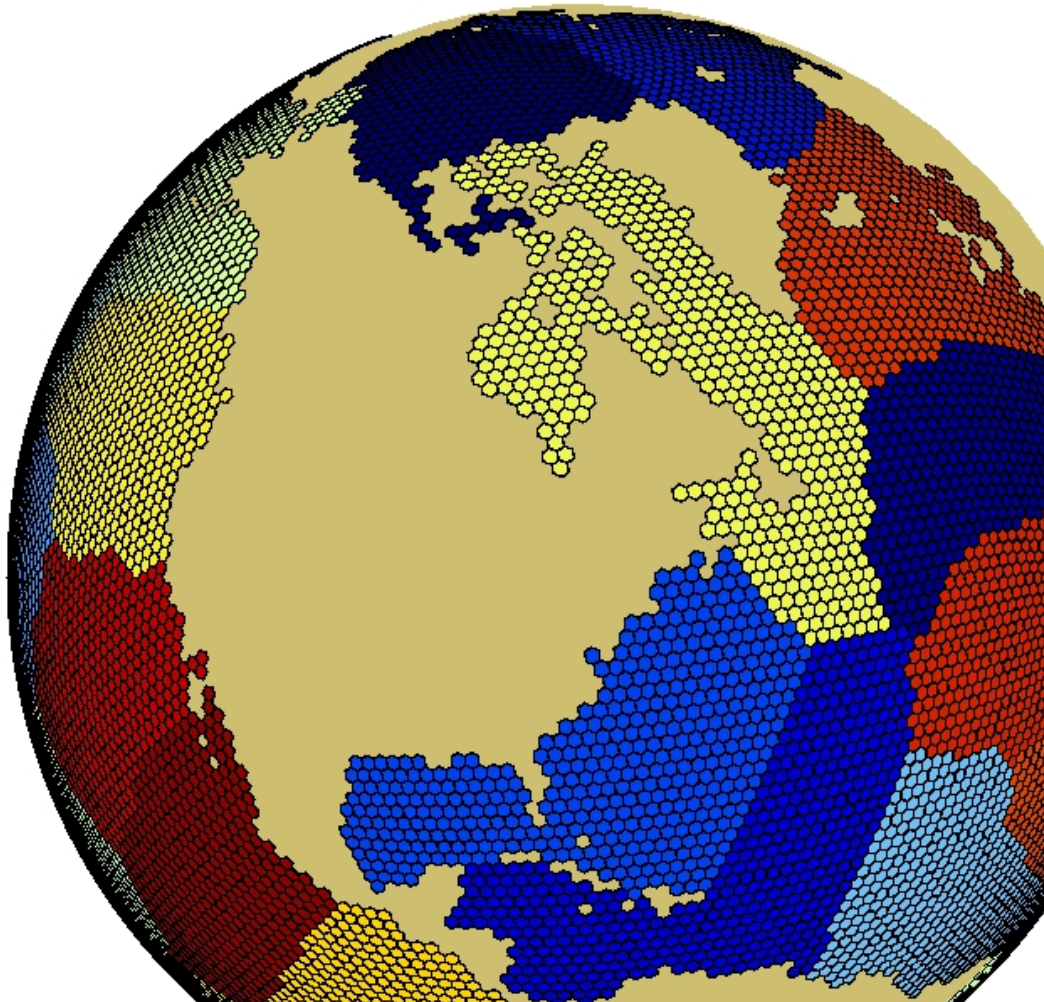
- By 2017
  - Improve FLOP performance to get near peak on a given HPC system.
  - Achieve high Simulated Year per Day (SYPD) performance on high-resolution grids (> 10 SYPD on 15km resolution on 3,000 cores).
  - Improve code scalability to utilize 100,000s cores.

3. MPAS-Ocean Computational Strategies

- Use unstructured, variable-resolution grids: Spherical Centriodal Voronoi Tessellation (SCVT).

- 15km resolution grid contains 2M cells, 4M vertices, 5.6M edges.

- MPI parallelism:
  - Decompose grid into blocks.
  - Construct multi-layered halos (typical 3 layers).
  - Communication: Halo exchanges with neighbor blocks, and all-to-all.



3. MPAS-Ocean Computational Strategies

Challenges:

- Unstructured grid means unstructured data resulting in inefficient data movement.
- Arbitrary ordering of blocks and cells causes low data locality and inefficient cache data reuse.
- Variable depth of cells (ocean depth) leads to load-imbalance.
- Multi-layered halos limit scaling –

  larger the number of processors, smaller the ratio of number of local cells to halo cells, and higher the communication costs.

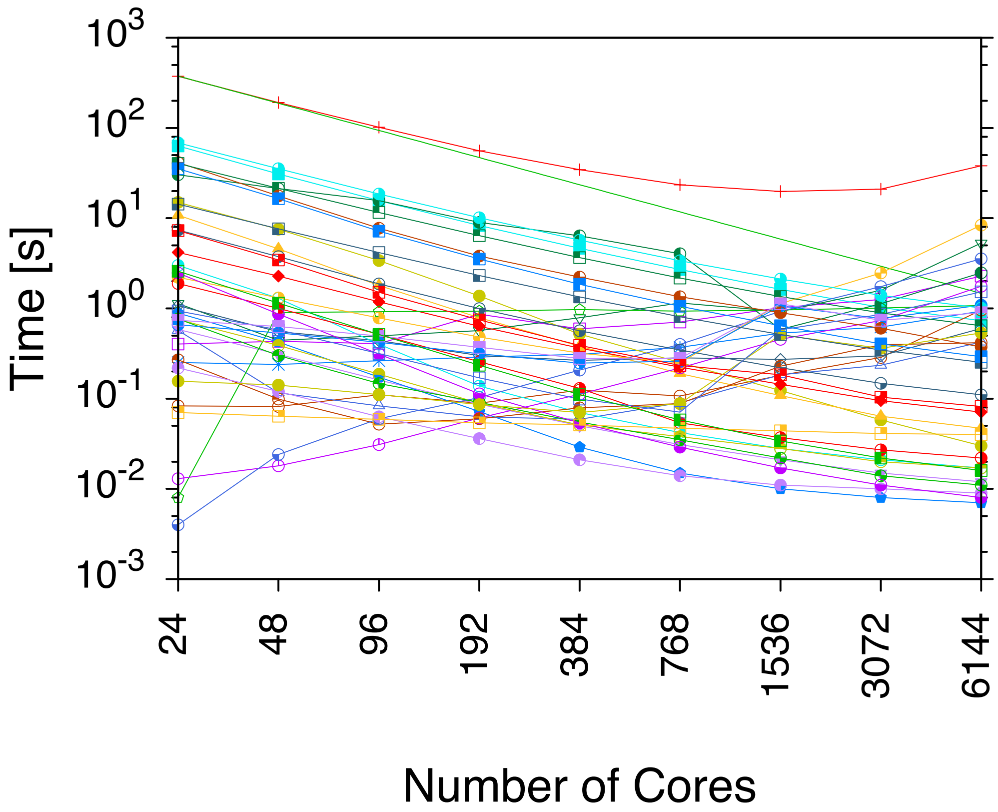
3. MPAS-Ocean Computational Strategies

Current performance optimization efforts:

- Intra-block cell and edge reordering based on space-filling curves (SFCs) to improve data locality and cache data usage, as well as blocks reordering to improve communication performance.
- Incorporation of OpenMP for multicore parallelization.
- Reorganization of data structures, and implementation of variable number of layers per cell (variable depth) to improve memory usage and data locality.
- Improved partitioning and load-balancing through weighted cells, and exploration of existing graph partitioning software, to minimize data transfers and communication.

4. Current HPC Usage

- MPAS-Ocean has been ported to Edison and Hopper at NERSC. Recently been ported to Titan (no GPU) at OLCF and Mira at ALCF.
- SUPER used 4.3M core-hours at NERSC in FY 2013 (asarje: 1.7M).
- Current simulations range up to 3K cores. Not scalable beyond!
- At most ~1 GB per core, ~24 GB per node on Hopper/Edison.
- Uses NetCDF for parallel I/O. Not I/O intensive.



4. Current HPC Usage

Performance Analysis Limitations:

- Few hardware counters available.
- No comprehensive cache usage statistics.
- No information on nodes’ allocation topology for a job.

5. HPC Requirements for 2017

- With higher usage anticipated for performance analysis, 10M core-hours needed per year.

- With improved parallelism, larger runs will also be analyzed for scaling up to 100,000s of cores.

- A typical run would use 10,000 cores, for several hours. Target is to achieve performance of more than 10 SYPD (Simulated Years Per Day) on 3,000 cores.

6. MPAS-Ocean Strategies for New Architectures

- Implementation on GPUs is planned for future: Offload compute kernel loops over grid cells and edges to GPUs.
- Incorporation of OpenMP threading support is currently underway.
- Efficient porting to Intel MIC is also planned. MPAS-Ocean currently runs natively on MIC but is not efficient.

Multiple institutes are involved through SUPER. Key institutes on MPAS-Ocean performance optimization efforts:

- Lawrence Berkeley National Lab (Leonid Oliker, Samuel Williams, Abhinav Sarje)
- Los Alamos National Lab (Douglas Jacobsen)
- Oak Ridge National Lab (Patrick Worley)
- University of Oregon (Kevin Huck)
- University of Utah (Mary Hall, Linda Wu)

7. High-Performance X-ray Scattering Data Analysis with *HipGISAXS*

- *HipGISAXS* (High-Performance Grazing Incidence Small Angle X-ray Scattering) *software suite* has been developed to meet computational needs at the Advanced Light Source (ALS).
- Simulation of scattering patterns for GISAXS.
- Discovery of nanostructures and nanoparticle system configurations through inverse modeling, using forward simulations (optimization problem).
- Computations on a structured 3-D grid.
- Each forward simulation is embarrassingly parallel – an ideal case for efficient exploitation of massive parallelism.

7. High-Performance X-ray Scattering Data Analysis with *HipGISAXS*

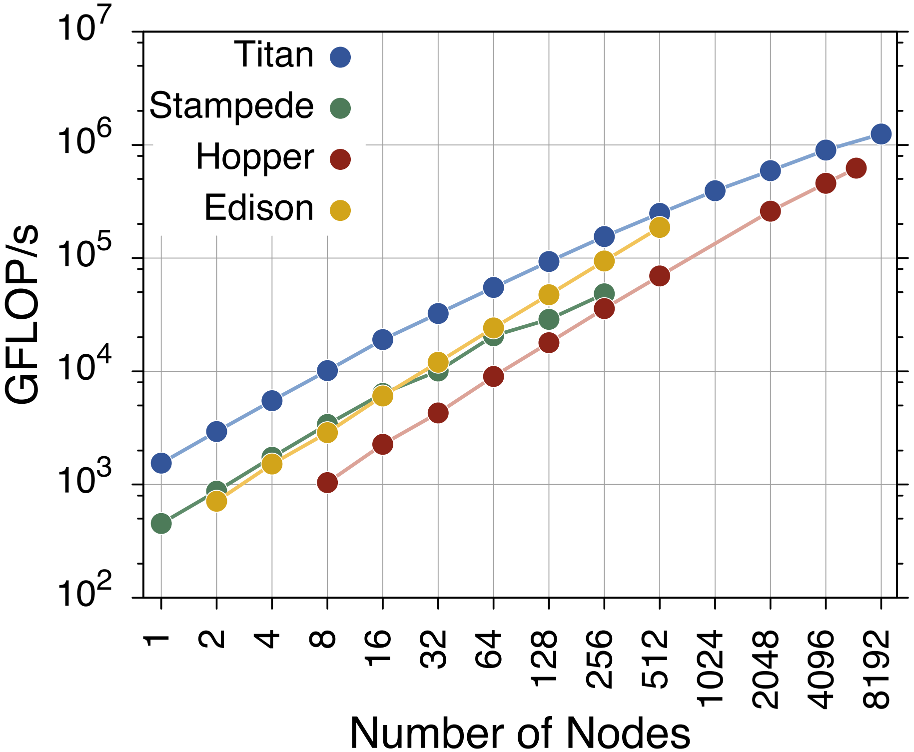
- Implements a hierarchical parallelization framework, dynamically decomposing computations on available resources for a perfect mapping to the hardware.
- Effectively exploits multicores, GPUs and Intel MICs at node-level.
- Performance improvements through architecture-aware optimizations and tuning.
- By 2017
  - Implement multiple optimization algorithms for structural modeling.
  - Achieve high FLOP rate during simulations close to the peak of the system – several PetaFLOPs on 100,000s of cores.
  - Implement a scattering data analysis pipeline to process data generated at beamlines at near real-time speed using NERSC's HPC resources.

8. HipGISAXS Computational Strategies

- Use structured 3-D grid.
- Typical grid resolution: 1000x1000x10, resulting in 10M cells.
- Custom shape definitions have 100,000s of triangles representing the shape surface: 10^{12} kernel computations in a single simulation.
- A single run may have 100s of forward simulations.
- MPI parallelism:
  - A hierarchy of MPI communicators for decomposing all independent computations efficiently.
  - Reduction operations performed at various levels.
- Node-level parallelism using OpenMP. Additional CUDA or Intel MIC parallelism (offload model) can be enabled based on available system resources. Vectorization at the lowest parallelism level.
- Planned efforts: Effective parallelization of inverse modeling algorithms.

9. Current HPC Usage

- HipGISAXS simulation has been ported to Edison and Hopper at NERSC, Titan at OLCF and Stampede at TACC.
- Performance analysis used 2.4M core-hours at NERSC in FY 2013.
- Current typical simulations range up to 3K cores. (But HipGISAXS scales well to 144,000 cores on Hopper).
- Memory usage minimization strategies result in minimal requirements per node.
- Uses HDF5 for parallel I/O. Not I/O intensive.



10. HPC Requirements for 2017

- With incorporation of inverse modeling anticipated, 10M core-hours needed per year.
- A typical run would use 10,000 cores, for several minutes.
- High-Performance processors with high degree of parallelism, such as GPUs, would be most performance and power efficient for HipGISAXS.
- Porting to other systems, such as Mira at ALCF, is also planned.

Current HipGISAXS development team at LBNL:

- Xiaoye Li (CRD)
- Slim Chourou (CRD)
- Abhinav Sarje (CRD)
- Alexander Hexemer (ALS)

11. Summary

- Faster and better climate and weather prediction with high-performance *MPAS-Ocean* code.
- Enable real-time X-ray scattering data analysis for nanostructure discovery in materials research with *HipGISAXS suite*.

- Recommend achieving a good balance between specialized high-efficiency manycores (beneficial for embarrassingly parallel computations) and general-purpose multicores (better resource utilization for unstructured computations).
- Better performance analysis tools needed. Accurate and easily accessible hardware performance counters necessary.
- Topology-aware nodes' allocation with accessible topology information would help in improving communication performance.